

A

Please type a plus sign (+) inside this box [+]

PTO/SB/05 (12/97)

Approved for use through 09/30/00. OMB 0651-0032

Patent and Trademark Office: U.S. DEPARTMENT OF COMMERCE

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

UTILITY PATENT APPLICATION TRANSMITTAL

(Only for new nonprovisional applications under 37 CFR 1.53(b))

Attorney Docket No. 04860.P2297

Total Pages 3

First Named Inventor or Application Identifier Thomas C. Clark

Express Mail Label No. EL164803317US

ADDRESS TO: **Assistant Commissioner for Patents
Box Patent Application
Washington, D. C. 20231**

APPLICATION ELEMENTS

See MPEP chapter 600 concerning utility patent application contents.

1. XX Fee Transmittal Form
(Submit an original, and a duplicate for fee processing)
2. XX Specification (Total Pages 21)
(preferred arrangement set forth below)
 - Descriptive Title of the Invention
 - Cross References to Related Applications
 - Statement Regarding Fed sponsored R & D
 - Reference to Microfiche Appendix
 - Background of the Invention
 - Brief Summary of the Invention
 - Brief Description of the Drawings (if filed)
 - Detailed Description
 - Claims
 - Abstract of the Disclosure
3. X Drawings(s) (35 USC 113) (Total Sheets 5)
4. X Oath or Declaration (Total Pages 4)
 - a. X Newly Executed (Original or Copy)
 - b. Copy from a Prior Application (37 CFR 1.63(d))
(for Continuation/Divisional with Box 17 completed) (**Note Box 5 below**)
 - i. DELETIONS OF INVENTOR(S) Signed statement attached deleting inventor(s) named in the prior application, see 37 CFR 1.63(d)(2) and 1.33(b).
5. Incorporation By Reference (useable if Box 4b is checked)
The entire disclosure of the prior application, from which a copy of the oath or declaration is supplied under Box 4b, is considered as being part of the disclosure of the accompanying application and is hereby incorporated by reference therein.
6. Microfiche Computer Program (Appendix)

UNITED STATES PATENT APPLICATION

FOR

USB Device Notification

Inventor: Thomas C. Clark

Prepared by:
BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN
12400 Wilshire Boulevard
Seventh Floor
Los Angeles, California 90025-1026
(408) 720-8598

Attorney's Docket No: 04860.P2297

"Express Mail" mailing label number: EL164803317US

Date of Deposit: 5-4-99

I hereby certify that I am causing this paper or fee to be deposited with the United States Postal Service "Express Mail Post Office to Addressee" service on the date indicated above and that this paper or fee has been addressed to the Assistant Commissioner for Patents, Washington, D. C. 20231

Barbara Herz
(Typed or printed name of person mailing paper or fee)

[Signature]
(Signature of person mailing paper or fee)

5-4-99
(Date signed)

USB Device Notification

BACKGROUND OF THE INVENTIONField of the Invention

5 The invention relates generally to communication in information processing systems and more specifically to notifying parts of systems that changes in other parts of the system have occurred.

Description of the Related Art

10 A recent advance in the design of computers involves hot-pluggability, the ability to connect a device to a computer bus while the computer is operating, and have that device communicate properly with the computer. In the past, computers could only be reliably connected to devices when power was not supplied. Now, busses have been designed which function properly when devices are connected or powered-up after the bus has power.

15 Unfortunately, solving the electrical problems associated with hot-pluggability does not guarantee that the device will communicate with the computer. The computer must be made aware that the device has been connected, and must be made aware of how that device communicates. This problem is typically solved either by having the computer poll the newly connected device for this information, or by having the newly
20 connected device announce its presence.

 Even this does not solve all problems involved with connecting a device to a computer after the computer starts operating. Clients, such as application programs and other routines may be established on the computer, as background processes, foreground processes, or as quiescent routines awaiting activation. If these clients have
25 already determined which devices are connected before the newly connected device is connected to the bus, the clients may not be able to take advantage of the presence of the newly connected device. Therefore, what is needed is a method or apparatus for

notifying clients of the presence of the newly connected device. Furthermore, not all application programs or routines may want to know of specific devices being connected, so what is also needed is a method or apparatus for selectively notifying clients of the presence of newly connected devices.

SUMMARY OF THE INVENTION

The present invention, in one embodiment, is a method of notifying clients of a change in a system including a client requesting notification of a change in the system, detecting the change in the system, and notifying the client requesting notification that the change in the system occurred. The present invention may further include maintaining a list of requests for notification and removing requests for notification when a client terminates a request for notification.

The invention, in an alternate embodiment, is a subsystem for notifying clients of a change in a system including means for a client to request notification of the change in the system, means for detecting the change in the system, and means for notifying the client requesting notification that the change in the system occurred.

The present invention may be implemented in a system including a processor and memory. Likewise, the present invention may be embodied in instructions encoded in a computer readable medium. Furthermore, the present invention may be practiced in a system employing a USB.

BRIEF DESCRIPTION OF THE DRAWINGS

The present invention is illustrated by way of example and not limitation in the accompanying figures.

Figure 1a illustrates a system suitable for use with the present invention.

5 Figure 1b illustrates an alternate conception of the system of Figure 1a.

Figure 2 illustrates another system suitable for use with the present invention.

Figure 3 illustrates a process followed in the practice of one embodiment of the present invention.

Figure 4 illustrates one representation of communication that may occur in practicing the present invention.

Figure 5 illustrates an example of a machine-readable medium in accordance with the present invention.

DETAILED DESCRIPTION

A method and apparatus for USB Device Notification is described. In the following description, for purposes of explanation, numerous specific details are set forth in order to provide a thorough understanding of the invention. It will be apparent, however, to one skilled in the art that the invention can be practiced without these specific details. In other instances, structures and devices are shown in block diagram form in order to avoid obscuring the invention.

Reference in the specification to "one embodiment" or "an embodiment" means that a particular feature, structure, or characteristic described in connection with the embodiment is included in at least one embodiment of the invention. The appearances of the phrase "in one embodiment" in various places in the specification are not necessarily all referring to the same embodiment.

Figure 1a illustrates a system suitable for use with the present invention. The system includes Host 100, HDD (Hard Disk Drive) 110, Keyboard 120, all connected to Bus 130. Additionally, Printer 140 may be connected to Bus 130, thereby allowing communication between Host 100 and Printer 140. Figure 1b illustrates an alternate conception of the system of Figure 1a. Here, the connections are formed with a star topology as typically used when making connections via the Universal Serial Bus (USB). There is a direct connection between Host 100 and Keyboard 120, between Host 100 and HDD 110, and between Host 100 and Hub 135. Printer 140 is optionally connected to Hub 135. In this example, the connections between the devices are the physical connections that substitute for the logical Bus 130. Further information on the USB may be obtained by consulting the USB Specification 1.0, January 15, 1996, which document is hereby incorporated by reference.

Host 100 may be a computer or other processing apparatus suitable for connection to peripheral devices. If a computer, Host 100 includes a microprocessor

and may also include some form of internal storage. However, Host 100 may be a controller which serves as a gateway for signals to and from the devices connected thereto.

Figure 2 illustrates another system suitable for use with the present invention.

5 Microprocessor 210 is coupled to Bus 230, which is coupled to both Memory 220 and USB Control 240. USB Control 240 is coupled to HDD 260, Modem 270 and Printer 280 through USB 250. In a typical USB implementation, USB Control 240 is a control hub that is directly connected to each of HDD 260, Modem 270, and Printer 280. Since the USB allows for hot insertion and removal, Modem 270 may be connected to USB Control 240 after Microprocessor 210 is powered, or Modem 270 may be disconnected from USB Control 240 while Microprocessor 210 is still powered.

Microprocessor 210 communicates with each of the devices coupled to USB Control 240, so Microprocessor 210 must be kept apprised of what is connected to USB Control 240. Otherwise, Microprocessor 210 may attempt to send instructions to Modem 270 after Modem 270 has been disconnected from USB Control 240. Likewise, Microprocessor 210 may need to access HDD 260. If HDD 260 was not connected to USB Control 240 when Microprocessor 210 initially sought access to HDD 260, Microprocessor 210 may fail to complete its processing due to its inability to access HDD 260.

20 As mentioned above, Host 100 may be a computer or other processing apparatus suitable for connection to peripheral devices. In one embodiment, Host 100 includes Microprocessor 210, Memory 220, Bus 230 and USB Control 240.

Microprocessor 210 may be a processor such as those manufactured by Motorola or Intel. Memory 220 may be dynamic random access memory (DRAM), and may also
25 include static RAM or ROM of some form. Note that Memory 220, Microprocessor 210 and Bus 230 may be incorporated on a single device, or each may be composed of several individual devices. Typically, Host 100 also includes some form of input device,

such as a keyboard, pointing device, or scanner, and some form of output device, such as a display or speaker. Host 100 may also be connected to input devices such as cameras, storage media, or microphones, and may also be connected to output devices such as printers or storage media. Host 100 may further be connected to such
5 input/output devices as modems or touch-sensitive screens, too. Storage media (machine-readable media) may include magnetic disks, carrier waves, optical disks, magnetic tape, memory as described with regard to Memory 220 above, and other forms of media suitable for storage of information.

It will be appreciated that Host 100 may take on other forms, such as a network
10 computer or intelligent appliance. Typically, Host 100 will be controlled by an operating system, such as MacOS available from Apple Computer Corp. of Cupertino, California, the Linux operating system, or Windows '95 available from Microsoft Corporation of Redmond, Washington. Under control of such an operating system, Host 100 will execute programs such as application programs. Likewise, the operating system will include routines. These routines and programs will, at times, access peripheral devices
15 coupled to Host 100. When these routines and programs access these peripheral devices, adequate information on which peripheral devices are currently coupled to Host 100 and what other changes have occurred in the system is essential.

These routines and programs may be thought of as clients for purposes of
20 discussion. A method of informing these clients of what changes have occurred in the system will now be explained. Figure 3 illustrates a process followed in the practice of one embodiment of the present invention, whereby the clients are notified of changes in the system. First, a client requests notification of certain changes in Request Entry 310. As a result of that request, Parameter Storage 320 occurs, storing parameters of the
25 request submitted by the client. Later, the client may no longer need information on changes, and therefore will terminate the previous request for notification at Request Termination 330. As a result of the termination, Parameter Removal 340 occurs, and

the parameters stored relating to the terminated request are removed. Requests may, in one embodiment, be submitted to a Device Notification routine, which will then either store or remove the parameters of the requests as appropriate.

Simultaneous with the operation on requests, the Device Notification routine will also determine whether changes are occurring in the system. When a change is detected as in Change Detection 360, the Device Notification routine acts upon that change. In Determination 370, the Device Notification routine will determine which requests have parameters that include the change detected in Change Detection 360. The Device Notification routine will then proceed to Notification 380, that is notifying each client that submitted a request with parameters that include the detected change. Both cycles Request Entry 310 through Parameter Removal 340 and Change Detection 360 through Notification 380 proceed independently of each other.

Figure 4 illustrates one representation of communication that may occur in practicing the present invention. Client 410 communicates with Device Notification routine 420, such as through requests for notification and termination of these requests. These requests may, in one embodiment, include an Installed Routine 430, which Device Notification 420 may activate in order to notify Client 410 of a detected change. Such an Installed Routine 430 may communicate with both Client 410 and Device Notification routine 420. Device Notification 420 communicates with Lower Level 440 to detect changes in the system, and then compares those changes with the requests for notification received from Client 410 to determine whether Client 410 needs notification of the detected changes. Lower Level 440 may be a part of an operating system, implemented in either hardware, software, or firmware. Alternatively, Lower Level 440 may be a physical device such as USB Control 240. Changes in the system may include connection or disconnection of a device such as a printer, errors in interfacing or communicating with a device, or a device being busy due to previous activity.

In one embodiment, utilized in implementing the present invention in conjunction with the MacOS operating system available from Apple Computer, Corp. of Cupertino, California, communication between a Client and a Device Notification routine occurs when the Client calls a USBInstallDeviceNotification function. Such a function call serves as a request for notification. A call to such a function looks like:

```
void USBInstallDeviceNotification(
    USBDeviceNotificationParameterBlock *pb);
```

10 pb A pointer to the USBDeviceNotificationParameterblock

The USBDeviceNotificationParameterblock includes information on what types of changes the Client wants notification of and what routine (Installed Routine) to use to communicate with the Client. The

USBDeviceNotificationParameterblock is defined as:

```
/* Device Notification Parameter Block */
struct USBDeviceNotificationParameterBlock
(
    UInt16                                      pbLength;
    UInt16                                      pbVersion;
    USBNotificationType                      usbDeviceNotification
    UInt8                                      reserved1;
    USBDeviceRef                              usbDeviceRef;
    UInt16                                      usbClass;
    UInt16                                      usbSubClass;
    UInt16                                      usbProtocol;
    UInt16                                      usbVendor
    UInt16                                      usbProduct
    OSStatus                                      result;
    UInt32                                      token;
    USBDeviceNotificationCallbackProcPtr      callback;
    UInt32                                      refcon;
};
```

35 Field descriptions

40	<--> usbDeviceNotification	The type of notification The following notifications are defined: kNotifyAnyEvent kNotifyAddDevice kNotifyAddInterface kNotifyRemoveDevice kNotifyRemoveInterface
----	----------------------------	---

	<code><-- usbDeviceRef</code>	The device reference for the target device
	<code><--> usbClass</code>	The class of the target device, use <code>kUSBAnyClass</code> for any class
5	<code><--> usbSubClass</code>	The Subclass of the target device, use <code>kUSBAnySubclass</code> for any subclass
10	<code><--> usbProtocol</code>	The protocol of the target device, use <code>kUSBAnyProtocol</code> for any protocol
	<code><--> usbVendor</code>	The Vendor ID of the target device, use <code>kUSBAnyVendor</code> for any vendor
15	<code><--> usbProduct</code>	The product ID of the target device, use <code>kUSBAnyProduct</code> for any product
20	<code><-- result</code>	The status of the call
	<code><-- token</code>	The identifier for this notification request
	<code>--> callback</code>	A pointer to the callback routine to be called when the notification criteria is satisfied

Note that an arrow '`-->`' indicates a parameter sent from the Client to the Device Notification routine, and an arrow '`<--`' indicates a parameter sent from the Device Notification routine to the Client. A double-ended arrow '`<-->`' indicates a parameter communicated in both directions. Also, note that the parameters sent from the Client to the Device Notification routine specify what type of changes the Client seeks notification of. A first request from a first Client might seek notification of the connection of any printer. A second request, from either a first Client or a second Client, may seek notification of connection of any printer manufactured by a particular vendor, such as Hewlett-Packard. The first request would simply specify a printer as the class of device in the `usbClass` field. The second request would specify a printer in the `usbClass` field, and also specify the vendor Hewlett-Packard in the `usbVendor` field.

As will be appreciated, requests for notification may be specific or general, and multiple requests may be made by a single Client, such as a Client seeking access to

both a modem and a printer. In one embodiment, the Device Notification routine provides notification by calling a callback routine such as the one supplied in the callback field. Such a routine would be declared as:

```

5 typedef void (USBDeviceNotificationCallbackProc)
      (USBDeviceNotificationParameterBlockPtr pb);
typedef USBDeviceNotificaitonCallbackProc
      *USBDeviceNotificationCallbackProcPtr;
10

```

This implementation provides flexibility to the Client, allowing the Client to receive notification of changes in a manner useful to the Client. In particular, the Client may supply a routine as a callback routine which alerts it to the presence of a newly connected device, or it may supply a routine that adds a newly connected device to a list of devices maintained by the Client. Likewise, notification of removal or disconnection of a device may result in calling a different routine supplied in a different request.

Since the Device Notification routine receives multiple requests, it must maintain a list of each request and its associated parameters. Such a list would include the parameters passed to the Device Notification routine when a client calls USBInstallDeviceNotification and a token identifying each request uniquely. However, if a Client such as an application program is terminating, the Client should terminate the request of notification, and it can do this with a call to a USBRemoveDeviceNotification function. Such a function is declared as:

```

25 OSStatus USBRemoveDeviceNotification (UInt32 token);
token          Notification identifier from the previously installed device
                notification routine.
30

```

Upon termination of the request, the information relating to that request that is maintained by the Device Notification routine is discarded, and the Device Notification routine no longer seeks to notify the Client in accordance with that request.

Turning to Figure 5, an example of a machine-readable medium in accordance with the present invention is illustrated. In one embodiment, Machine-Readable Medium 500 contains Operating System routines 520 embodying Device Notification 420 of Figure 4, Client 510, the routines that embody Client 410 of Figure 4, List of Requested Notices 530, and Callback routine 540, embodying Installed Routine 430 of Figure 4. Client 510 requests notification of changes in the status of the system. Operating System routines 520, such as those in the MacOS operating system available from Apple Computer, Inc. of Cupertino California, receive the requests for notification from Client 510, manage the List of Requested Notices 530 which stores the information on each request, detect changes in the status of the system, and call the Callback routine 540 when a status change corresponding to a request is detected or otherwise notify a requestor of the status change. List of Requested Notices 530 includes information on each request, such as that described in the above description of the invention as implemented in the MacOS operating system, and may include a pointer to Callback routine 540 or a similar callback routine for each individual request. Callback routine 540 calls Client 510 or otherwise notifies Client 510 of the change in status, and may be a portion of Client 510.

It will be appreciated that each of the above routines or portions of information may be stored in machine-readable media (or a single medium) in distributed or whole form. In either case, the information will typically be stored in a form suitable for execution (such as executable instructions for example) by a processor such as a microprocessor, or for use during execution by a processor. Additionally, the information, such as the List of Requested Notices 530, may be changed during execution, including creation or deletion of entries or creation or deletion of the entire List 530. In Figure 5, the appearance is created that everything is stored in integrated or whole form in a single machine-readable medium, but it will be appreciated that

storage in distributed form in one medium or over multiple media does not depart from the spirit of the invention.

In the foregoing detailed description, the method and apparatus of the present invention has been described with reference to specific exemplary embodiments thereof. It will, however, be evident that various modifications and changes may be made thereto without departing from the broader spirit and scope of the present invention. The present specification and figures are accordingly to be regarded as illustrative rather than restrictive.

CLAIMS

What is claimed is:

1 1. A method of notifying clients of a change in a system comprising:
2 a client requesting notification of the change in the system;
3 detecting the change in the system; and
4 notifying the client requesting notification that the change in the system occurred.

1 2. The method of claim 1 further comprising:
2 maintaining a list of requests for notification.

1 3. The method of claim 1 further comprising:
2 the client terminating a request for notification.

1 4. The method of claim 2 further comprising:
2 the client terminating a request for notification;
3 and removing a request corresponding to the client from the list of requests for
4 notification.

1 5. The method of claim 1 wherein:
2 the change in the system is connection of a device.

1 6. The method of claim 1 wherein:
2 the change in the system is disconnection of a device.

1 7. The method of claim 1 wherein:

2 said requesting includes the client supplying a callback routine; and

3 said notifying includes executing the callback routine.

1 8. A subsystem for notifying clients of a change in a system comprising:

2 means for a client to request notification of the change in the system;

3 means for detecting the change in the system; and

4 means for notifying the client requesting notification that the change in the
5 system occurred.

1 9. The subsystem of claim 8 further comprising:

2 means for maintaining a list of requests for notification.

1 10. The subsystem of claim 9 further comprising:

2 means for the client to terminate a request for notification; and

3 means for removing a request corresponding to the client from the list of requests
4 for notification.

1 11. The subsystem of claim 10 further comprising:

2 means for communication to the client; and

3 wherein:

4 the client supplies the means for communication; and

5 the means for communication is utilized by the means for notifying.

12. A machine-readable medium containing a plurality of executable instructions,
which when executed on a processor cause said processor to perform a method of
notifying clients of a change in a system, the method comprising:

a client requesting notification of the change in the system;
detecting the change in the system; and
notifying the client requesting notification that the change in the system occurred.

13. The machine-readable medium of claim 12 wherein the method further
comprises:

maintaining a list of requests for notification.

14. The machine-readable medium of claim 13 wherein the method further
comprises:

the client terminating a request for notification;
and removing a request corresponding to the client from the list of requests for
notification.

15. The machine-readable medium of claim 14 wherein:
said requesting includes the client supplying a callback routine; and
said notifying includes executing the callback routine.

1 16. A system comprising:
2 a processor;
3 a memory;
4 a bus, the bus coupled to the processor, the bus coupled to the memory; and
5 the processor processing a request by a client for notification of a change in the
6 system, the processor detecting the change in the system, and the processor notifying
7 the client that the change in the system has occurred.

1 17. The system of claim 16 wherein:
2 the processor maintains a list of requests for notification.

1 18. The system of claim 17 wherein:
2 the processor stores the list of requests in memory.

1 19. The system of claim 17 wherein:
2 the processor processes the client's termination of a request for notification by
3 removing a request corresponding to the client from the list of requests for notification.

1 20. The system of claim 19 wherein:
2 the processor receives a callback routine from the client when the client requests
3 notification and the processor notifies the client by executing the callback routine.

1 21. A method of notifying clients of a change in a USB comprising:
2 a first client requesting notification of a first change in the USB;
3 detecting the first change in the USB; and
4 notifying the first client requesting notification that the first change in the USB
5 occurred.

1 22. The method of claim 21 wherein:
2 the change is connection of a device to the USB;
3 and further comprising:
4 finding a driver suitable for use with the device.

1 23. The method of claim 21 wherein:
2 the change is disconnection of a device from the USB;
3 and further comprising:
4 deactivating a driver corresponding to the device.

1 24. The method of claim 21 further comprising:
2 a second client requesting notification of a second change in the USB;
3 detecting the second change in the USB; and
4 notifying the second client requesting notification that the second change in the
5 USB occurred.

1 25. The method of claim 24 wherein:
2 a change in the USB constitutes a first change and constitutes a second change.

- 1 26. The method of claim 24 wherein:
- 2 a change in the USB that constitutes a first change does not constitute a second
- 3 change.

ABSTRACT OF THE DISCLOSURE

A method of notifying clients of a change in a USB including a first client requesting notification of a first change in the USB, detecting the first change in the USB, and notifying the first client requesting notification that the first change in the USB occurred.

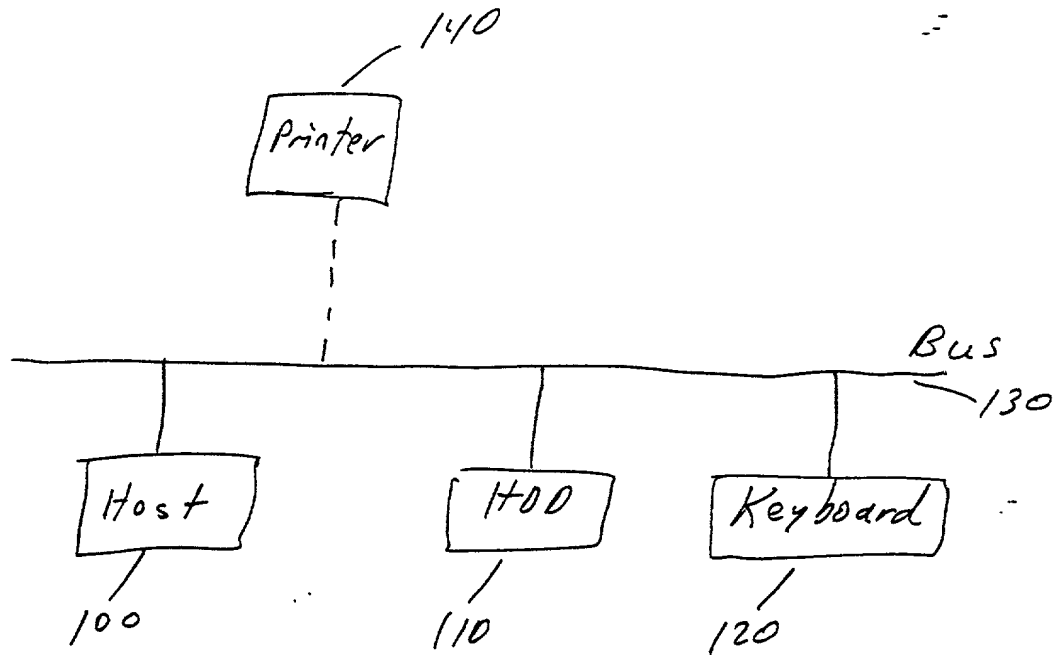


Fig. 1a

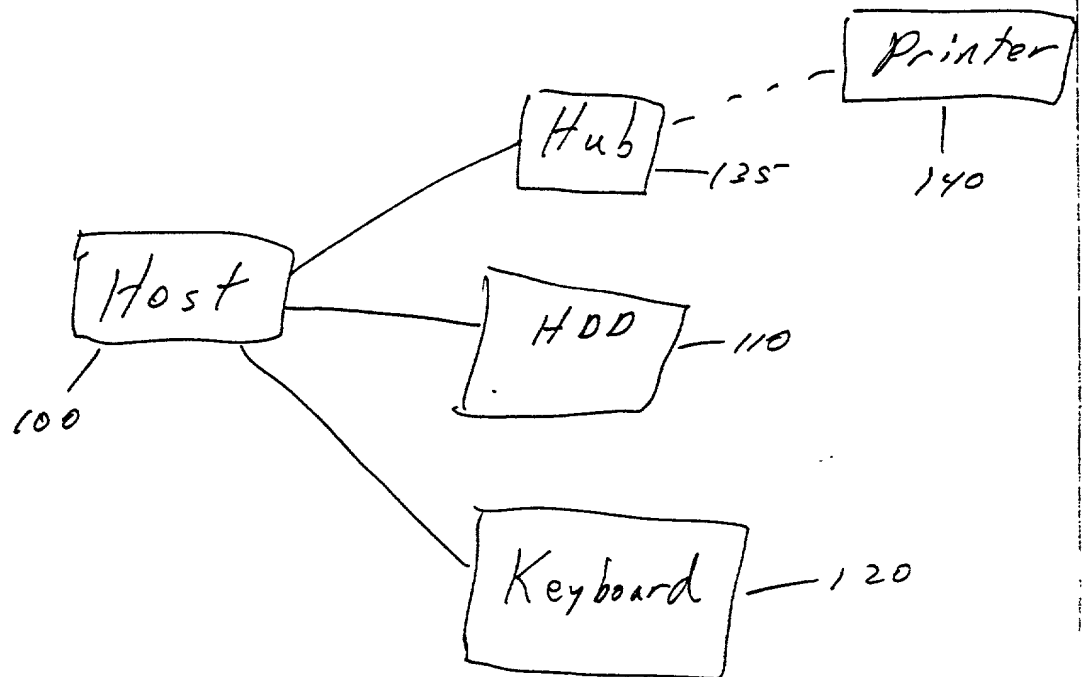


Fig. 1b

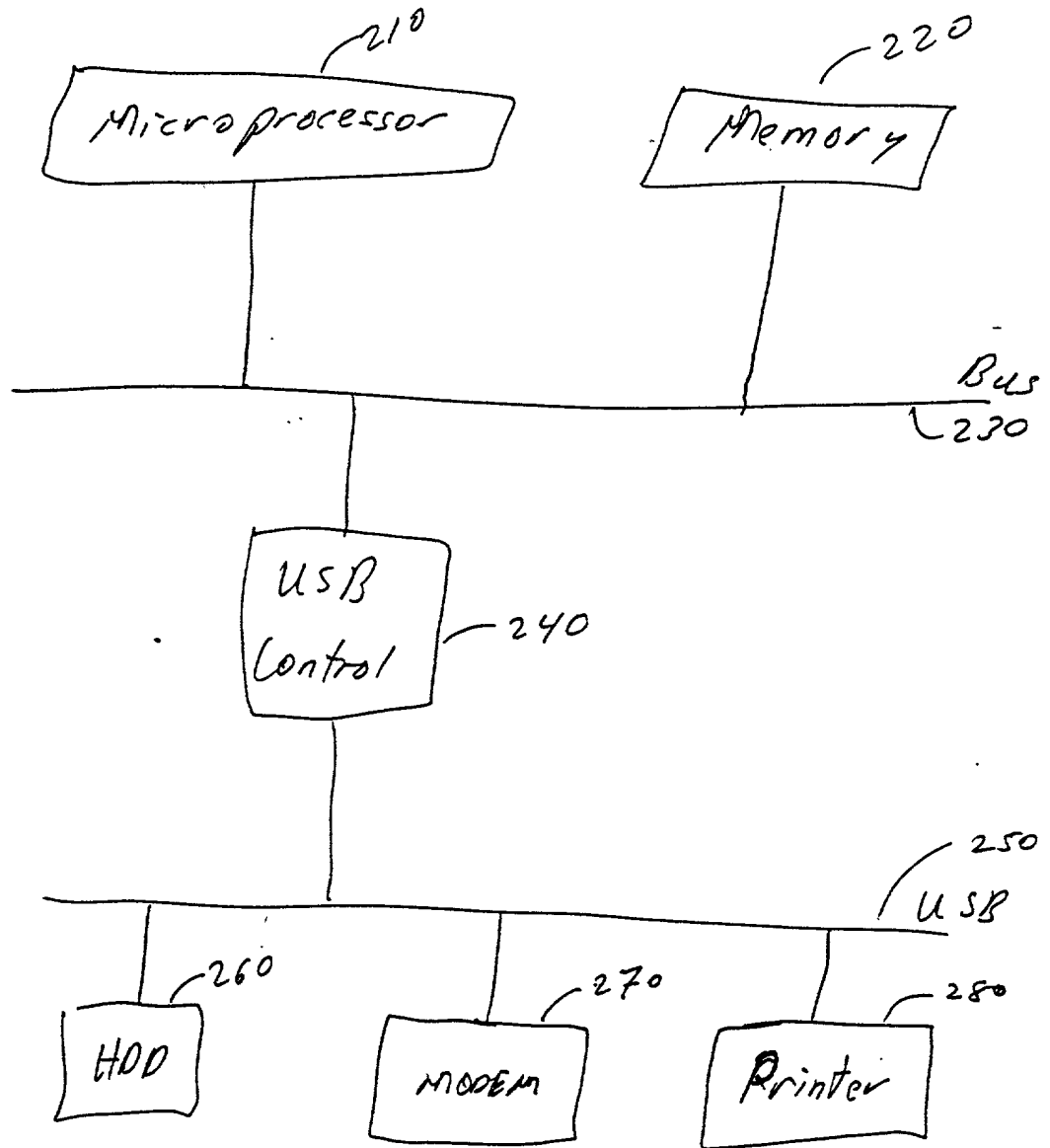


Fig. 2

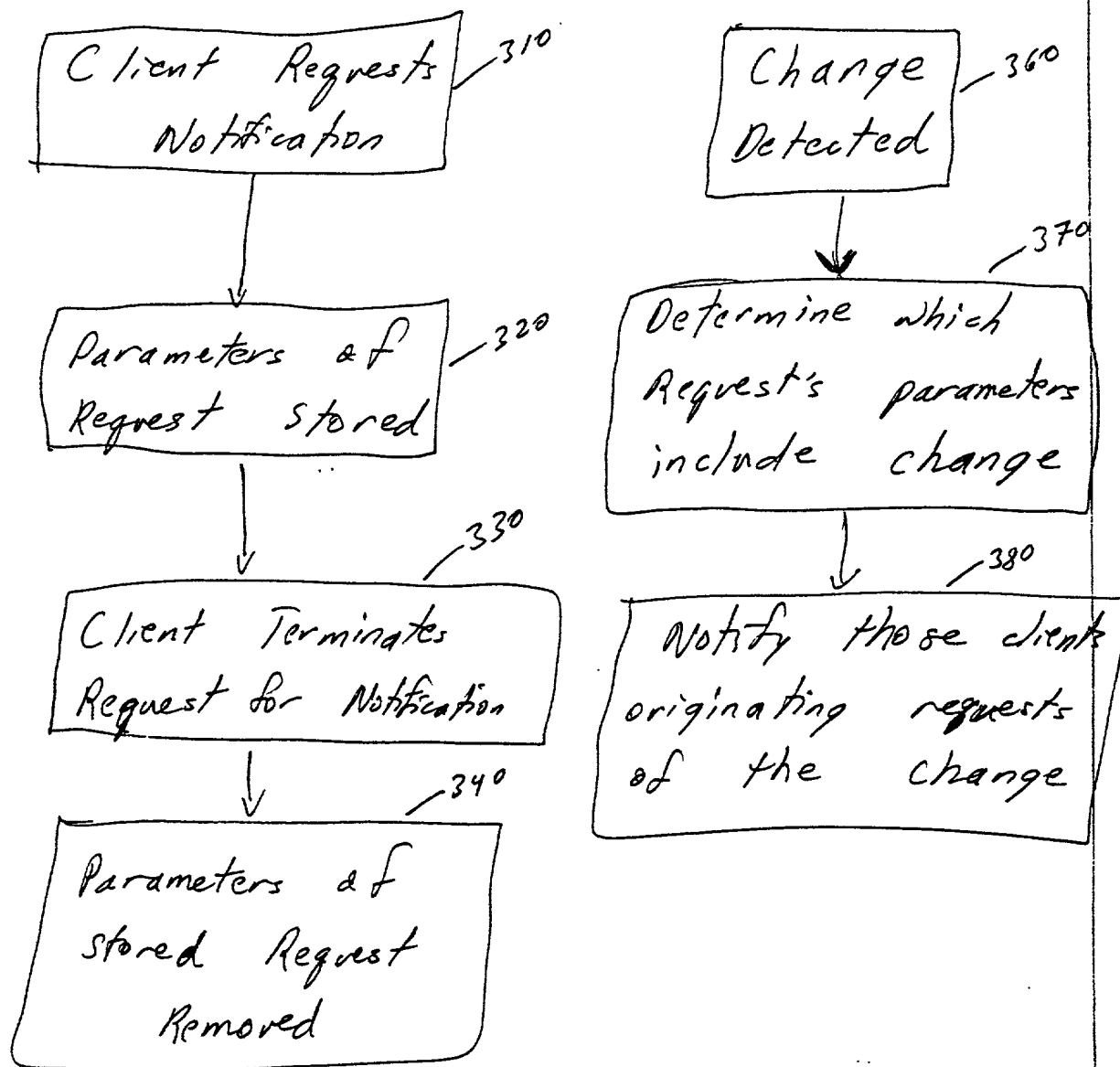


Fig. 3

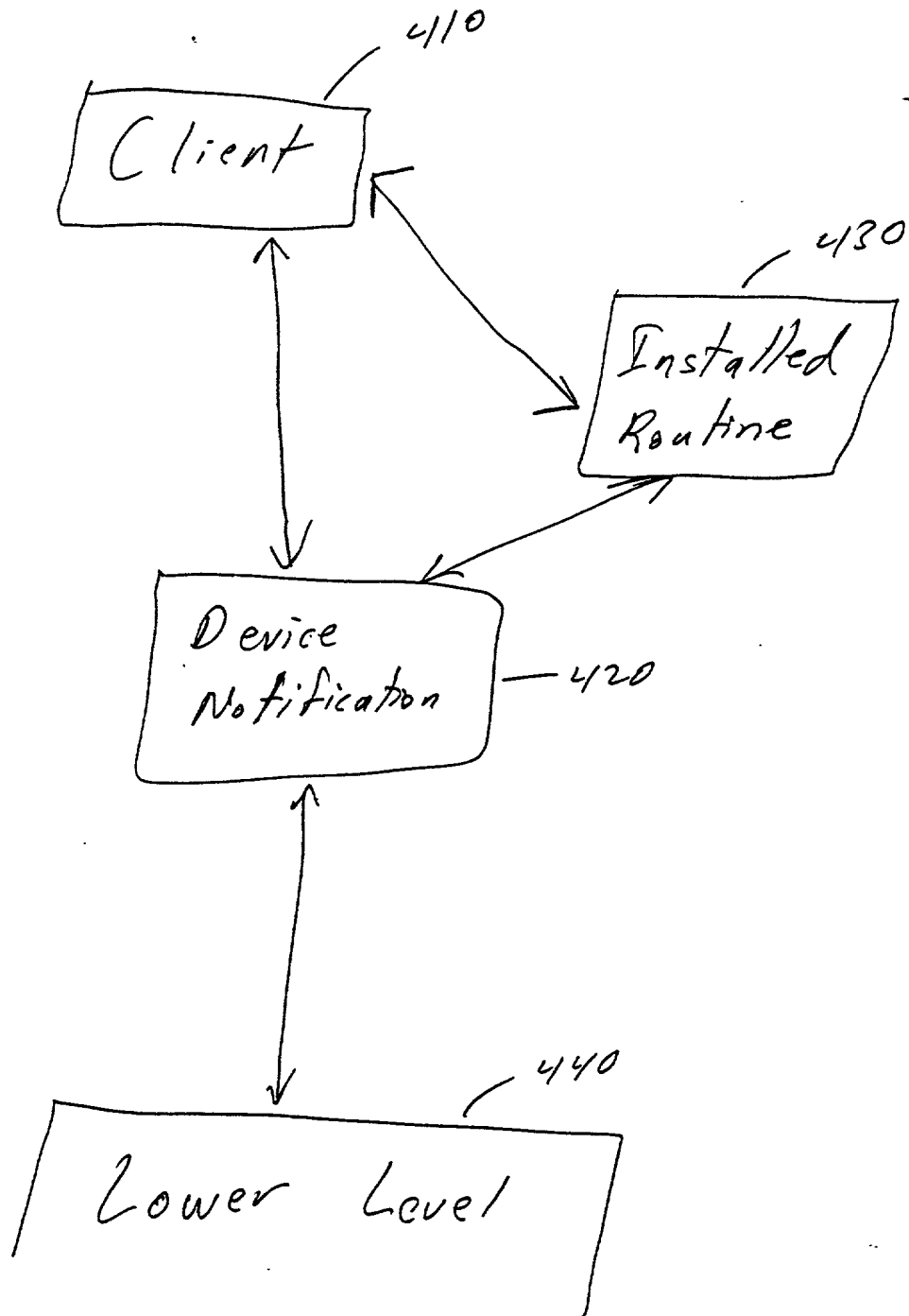


Fig. 4

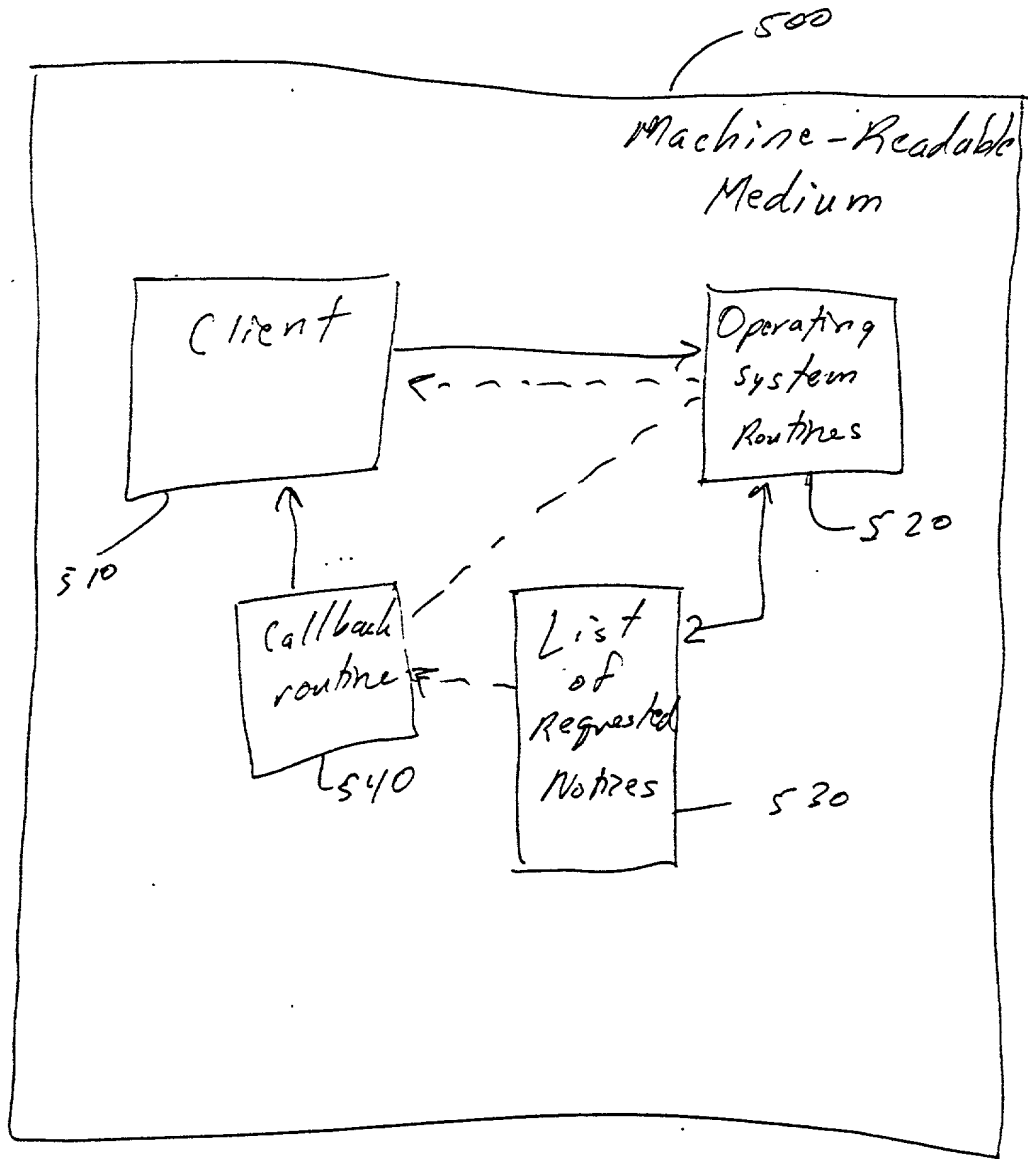


Fig. 5

DECLARATION AND POWER OF ATTORNEY FOR PATENT APPLICATION

As a below named inventor, I hereby declare that:

My residence, post office address and citizenship are as stated below, next to my name.

I believe I am the original, first, and sole inventor (if only one name is listed below) or an original, first, and joint inventor (if plural names are listed below) of the subject matter which is claimed and for which a patent is sought on the invention entitled

USB DEVICE NOTIFICATION

the specification of which

XX is attached hereto.
_____ was filed on _____ as
United States Application Number _____
or PCT International Application Number _____
and was amended on _____
(if applicable)

I hereby state that I have reviewed and understand the contents of the above-identified specification, including the claim(s), as amended by any amendment referred to above. I do not know and do not believe that the claimed invention was ever known or used in the United States of America before my invention thereof, or patented or described in any printed publication in any country before my invention thereof or more than one year prior to this application, that the same was not in public use or on sale in the United States of America more than one year prior to this application, and that the invention has not been patented or made the subject of an inventor's certificate issued before the date of this application in any country foreign to the United States of America on an application filed by me or my legal representatives or assigns more than twelve months (for a utility patent application) or six months (for a design patent application) prior to this application.

I acknowledge the duty to disclose all information known to me to be material to patentability as defined in Title 37, Code of Federal Regulations, Section 1.56.

I hereby claim foreign priority benefits under Title 35, United States Code, Section 119(a)-(d), of any foreign application(s) for patent or inventor's certificate listed below and have also identified below any foreign application for patent or inventor's certificate having a filing date before that of the application on which priority is claimed:

<u>Prior Foreign Application(s)</u>			<u>Priority Claimed</u>	
<u>(Number)</u>	<u>(Country)</u>	<u>(Day/Month/Year Filed)</u>	<u>Yes</u>	<u>No</u>
<u>(Number)</u>	<u>(Country)</u>	<u>(Day/Month/Year Filed)</u>	<u>Yes</u>	<u>No</u>

I hereby claim the benefit under title 35, United States Code, Section 119(e) of any United States provisional application(s) listed below

(Application Number)	Filing Date
(Application Number)	Filing Date

I hereby claim the benefit under Title 35, United States Code, Section 120 of any United States application(s) listed below and, insofar as the subject matter of each of the claims of this application is not disclosed in the prior United States application in the manner provided by the first paragraph of Title 35, United States Code, Section 112, I acknowledge the duty to disclose all information known to me to be material to patentability as defined in Title 37, Code of Federal Regulations, Section 1.56 which became available between the filing date of the prior application and the national or PCT international filing date of this application:

(Application Number)	Filing Date	(Status -- patented, pending, abandoned)
(Application Number)	Filing Date	(Status -- patented, pending, abandoned)

I hereby appoint Farzad E. Amini, Reg. No. P42,261; Aloysius T. C. AuYeung, Reg. No. 35,432; Amy M. Armstrong, Reg. No. 42,265; William Thomas Babbitt, Reg. No. 39,591; Carol F. Barry, Reg. No. 41,600; Jordan Michael Becker, Reg. No. 39,602; Bradley J. Berezna, Reg. No. 33,474; Michael A. Bernadicou, Reg. No. 35,934; Roger W. Blakely, Jr., Reg. No. 25,831; Gregory D. Caldwell, Reg. No. 39,926; Kent M. Chen, Reg. No. 39,630; Lawrence M. Cho, Reg. No. 39,942; Yong S. Choi, Reg. No. P43,324; Thomas M. Coester, Reg. No. 39,637; Roland B. Cortes, Reg. No. 39,152; Barbara Bokanov Courtney, Reg. No. 42,442; Michael Anthony DeSanctis, Reg. No. 39,957; Daniel M. De Vos, Reg. No. 37,813; Robert Andrew Diehl, Reg. No. 40,992; Tarek N. Fahmi, Reg. No. 41,402; James Y. Go, Reg. No. 40,621; Richard Leon Gregory, Jr., Reg. No. 42,607; Dinu Gruia, Reg. No. P42,996; David R. Halvorson, Reg. No. 33,395; Thomas A. Hassing, Reg. No. 36,159; Phuong-Quan Hoang, Reg. No. 41,839; Willmore F. Holbrow III, Reg. No. P41,845; George W. Hoover II, Reg. No. 32,992; Eric S. Hyman, Reg. No. 30,139; Dag H. Johansen, Reg. No. 36,172; William W. Kidd, Reg. No. 31,772; Michael J. Mallie, Reg. No. 36,591; Andre L. Marais, under 37 C.F.R. § 10.9(b); Paul A. Mendonsa, Reg. No. 42,879; Darren J. Milliken, Reg. No. 42,004; Thinh V. Nguyen, Reg. No. 42,034; Kimberley G. Nobles, Reg. No. 38,255; Michael A. Proksch, Reg. No. 43,021; Babak Redjaian, Reg. No. 42,096; James H. Salter, Reg. No. 35,668; William W. Schaal, Reg. No. 39,018; James C. Scheller, Reg. No. 31,195; Anand Sethuraman, Reg. No. P43,351; Charles E. Shemwell, Reg. No. 40,171; Maria McCormack Sobrino, Reg. No. 31,639; Stanley W. Sokoloff, Reg. No. 25,128; Allan T. Sponseller, Reg. No. 38,318; Judith A. Szepesi, Reg. No. 39,393; Vincent P. Tassinari, Reg. No. 42,179; Edwin H. Taylor, Reg. No. 25,129; George G. C. Tseng, Reg. No. 41,355; Lester J. Vincent, Reg. No. 31,460; Glenn E. Von Tersch, Reg. No. 41,364; John Patrick Ward, Reg. No. 40,216; Stephen Warhola, Reg. No. 43,237; Charles T. J. Weigell, Reg. No. 43,398; Ben J. Yorks, Reg. No. 33,609; and Norman Zafman, Reg. No. 26,250; my attorneys, and James A. Henry, Reg. No. 41,064; Daniel E. Ovanezian, Reg. No. 41,236; and Chad R. Walsh, Reg. No. 43,235; my patent agents, of BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN LLP, with offices located at 12400 Wilshire Boulevard, 7th Floor, Los Angeles, California 90025, telephone (310) 207-3800, and James R. Thein, Reg. No. 31,710, my patent attorney; with full power of substitution and revocation, to prosecute this application and to transact all business in the Patent and Trademark Office connected herewith. I also hereby appoint Albert P. Cefalo, Reg. No. 27,315; Mark Aaker, Reg. No. 32,667; Richard Liu, Reg.

No. 34,377; Helene Plotka Workman, Reg. No. 35,981; Edward W. Scott, IV, Reg. No. 36,000; and Nancy R. Simon, Reg. No. 36,930; my attorneys; of APPLE COMPUTER, INC., located at 1 Infinite Loop, MS: 38-PAT, Cupertino, California 95014, telephone (408)974-9453, will full power of substitution and revocation, to prosecute this application and to transact all business in the Patent and Trademark Office connected herewith.

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

Full Name of Sole/First Inventor Thomas C. Clark

Inventor's Signature Thomas C. Clark Date 5/4/99

Residence San Jose, California Citizenship USA
(City, State) (Country)

Post Office Address 2094 Bello Avenue
San Jose, California 95125

Full Name of Second/Joint Inventor _____

Inventor's Signature _____ Date _____

Residence _____ Citizenship _____
(City, State) (Country)

Post Office Address _____

Full Name of Third/Joint Inventor _____

Inventor's Signature _____ Date _____

Residence _____ Citizenship _____
(City, State) (Country)

Post Office Address _____

Title 37, Code of Federal Regulations, Section 1.56
Duty to Disclose Information Material to Patentability

(a) A patent by its very nature is affected with a public interest. The public interest is best served, and the most effective patent examination occurs when, at the time an application is being examined, the Office is aware of and evaluates the teachings of all information material to patentability. Each individual associated with the filing and prosecution of a patent application has a duty of candor and good faith in dealing with the Office, which includes a duty to disclose to the Office all information known to that individual to be material to patentability as defined in this section. The duty to disclosure information exists with respect to each pending claim until the claim is cancelled or withdrawn from consideration, or the application becomes abandoned. Information material to the patentability of a claim that is cancelled or withdrawn from consideration need not be submitted if the information is not material to the patentability of any claim remaining under consideration in the application. There is no duty to submit information which is not material to the patentability of any existing claim. The duty to disclosure all information known to be material to patentability is deemed to be satisfied if all information known to be material to patentability of any claim issued in a patent was cited by the Office or submitted to the Office in the manner prescribed by §§1.97(b)-(d) and 1.98. However, no patent will be granted on an application in connection with which fraud on the Office was practiced or attempted or the duty of disclosure was violated through bad faith or intentional misconduct. The Office encourages applicants to carefully examine:

(1) Prior art cited in search reports of a foreign patent office in a counterpart application, and

(2) The closest information over which individuals associated with the filing or prosecution of a patent application believe any pending claim patentably defines, to make sure that any material information contained therein is disclosed to the Office.

(b) Under this section, information is material to patentability when it is not cumulative to information already of record or being made of record in the application, and

(1) It establishes, by itself or in combination with other information, a prima facie case of unpatentability of a claim; or

(2) It refutes, or is inconsistent with, a position the applicant takes in:

(i) Opposing an argument of unpatentability relied on by the Office, or

(ii) Asserting an argument of patentability.

A prima facie case of unpatentability is established when the information compels a conclusion that a claim is unpatentable under the preponderance of evidence, burden-of-proof standard, giving each term in the claim its broadest reasonable construction consistent with the specification, and before any consideration is given to evidence which may be submitted in an attempt to establish a contrary conclusion of patentability.

(c) Individuals associated with the filing or prosecution of a patent application within the meaning of this section are:

(1) Each inventor named in the application;

(2) Each attorney or agent who prepares or prosecutes the application; and

(3) Every other person who is substantively involved in the preparation or prosecution of the application and who is associated with the inventor, with the assignee or with anyone to whom there is an obligation to assign the application.

(d) Individuals other than the attorney, agent or inventor may comply with this section by disclosing information to the attorney, agent, or inventor.